Network Working Group
INTERNET-DRAFT
Category: Informational
draft-lior-radius-prepaid-extensions-05.txt
Expires: 17 January, 2005

A. Lior
Bridgewater Systems
P. Yegani
Cisco
K. Chowdhury
Nortel
Y. Li
Bridgewater Systems
July 19, 2004

PrePaid Extensions to Remote Authentication Dial-In User Service (RADIUS)

Status of this Memo

Copyright Notice

Abstract

The draft presents an extension to the Remote Authentication Dial-In User Service (RADIUS) protocol to support PrePaid data services for a wide range of deployments such as Dial, Wireless, WLAN. Consideration for roaming using mobile-ip is also given.

Table of Contents

1. Introduction

This draft describes RADIUS protocol extensions supporting PrePaid Data Services.

PrePaid data services are cropping up in many wireless and wireline based networks. A PrePaid Data Service subscriber is one that purchases a contract to receive a data service for either a period of time, or a quantity of data. Before providing a prepaid data service, the service provider checks that the prepaid subscriber has sufficient funds to cover the particular service request. Only after confirmation that funds are available is the service provided to the user.

The subscriber purchases the Data Service using various means such as buying a PrePaid Card, or online. How the subscriber purchases their PrePaid Data Service depends on the deployment and is not in scope for this document.

In some deployments, the PrePaid data service will be combined with other Prepaid services such as PrePaid circuit voice service. This is not an issue for this document other than the fact that the PrePaid Data Services described in this paper should work with other PrePaid data and or circuit voice services.

The fundamental business driver for a carrier to provide PrePaid data services is to increase participation (subscriber base) and thus to increase revenues. Therefore, it makes sense that PrePaid services meet the following goals:

- Leverage existing infrastructure, hence reducing capital expenditures typically required when rolling out a new service;
- Ability to rate service requests in real-time;
- Ability to check that the end userÆs account for coverage for the requested service charge prior to execution of that service;
- Protect against revenue loss, i.e., prevent an end user from generating chargeable events when the credit of that account is exhausted or expired;
- Protect against fraud;
- Be as widely deployable over Dialup, Wireless and WLAN networks.

The protocol described in this document maximizes existing infrastructure as much as possible û hence the use of the RADIUS protocol. The protocol is used in ways to protect against revenue loss or revenue leakage. This is achieved by defining procedures for the real-time delivery of service information to a pre-paid enabled AAA server, to minimize the financial risk, for the pre-paid enabled AAA server to be able to allocate small quotas to each data session and having the ability to update the quotas from a central quota server dynamically during the lifetime of the PrePaid data session. As well, mechanisms have been designed to be able to recover from errors that occur from time to time.

Protection against fraud is provided by recording of accounting records, by providing mechanisms to thwart replay attacks. As well, mechanisms have been provided to terminate data sessions when fraud is detected.

PrePaid System will become more prevalent and sophisticated as the various networks such as Dialup, Wireless and WLAN converge. This protocol extension is designed to meet the challenges of converged networks. The draft mainly addresses how to use the RADIUS protocol to achieve a PrePaid Data Service. The prepaid architecture assumes that rating of chargeable events does not occur in the element providing the service. This rating could be performed in the prepaid enabled AAA server or may exist in an entity behind this AAA server. Business logic and service rules may define that tariffing of events vary in time, e.g., the particular price per megabyte download may be defined to switch at 8pm from a high tariff to a low tariff. The RADIUS extensions for prepaid support scenarios enable scalable implementation of tariff switched prepaid systems.

Furthermore, the prepaid architecture assumes that a quota server is available which, through co-ordination with the rating entity and centralized balance manager is able to provide a quota response in response for prepaid data service. This quota server functionality could be performed in the prepaid enabled AAA server or may exist in an entity behind this AAA server. Finally, the details of the PrePaid System, such as its persistent store, how it maintains its accounts are not covered at all. However, in order to define the RADIUS protocol extensions it is necessary to discuss the functional behavior of the PrePaid System.

1.1 Terminology

Service Access Device	
PrePaid Client(PPC)	
PrePaid Server(PPS)	
Home agent (HA)	
Home network	
Home AAA (HAAA)	
Broker AAA (BAAA)	
Visited AAA (VAAA)	
Foreign Agent (FA)	
WLAN	
Service Event	
Access Service	The service that is provided to the user when the user is authenticated and authorized. In this document the term is used to differentiate between authorization of services that are explicitly identified by a Service Id. Example of Access Service would be the Main Service instance of 3GPP2.

1.2 Requirements language

In this document, several words are used to signify the requirements of the specification. These words are often capitalized. The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

2. Architectural Model

The architectural model supports prepaid clients on a service access device. A service access device (e.g. a NAS) typically provides a access to data service to end-users. A service access device in an entity on the data path that includes a RADIUS client.

When pre-paid service is used the service access device collects service event information and reports it while and/or after services are provided to the prepaid user. This event information is sent to a prepaid server by using the prepaid RADIUS extensions.

If real-time credit control is required, the service access device (prepaid client) contacts the prepaid server with service event

information included before the service is provided. The prepaid server, depending on the service event information, performs credit check and allocates a portion of available credit to the service event. The rating entity converts this credit value into a time and/or volume amount, which is then returned to the requesting service access device. The rating entity may determine that during the allocated quota, a tariff switch will occur in which case the rating entity will include details of the quota allocated prior to the tariff switch, details of the quota allocated after the tariff switch together with details of when the tariff switch will occur.

The requesting service access device then monitors service execution according to the instructions returned by the prepaid server. After service completion or on a subsequent request for service, the prepaid server deducts the reserved allocation of credit from the prepaid userÆs account.

Similarly, when a user terminates an on-going prepaid service, the prepaid client signals the prepaid server with the a value corresponding to the unused portion of the allocated quota. The prepaid server is then able to refund unused allocated funds into a userÆs prepaid account.

There MAY be multiple prepaid servers in the system for reasons of redundancy and load balancing. The system MAY also contain separate rating server(s) and accounts MAY be located in a centralized database. System internal interfaces can exist to relay messages between servers and an account manager. However the detailed architecture of prepaid system and its interfaces are implementation specific and are out of scope of this specification.

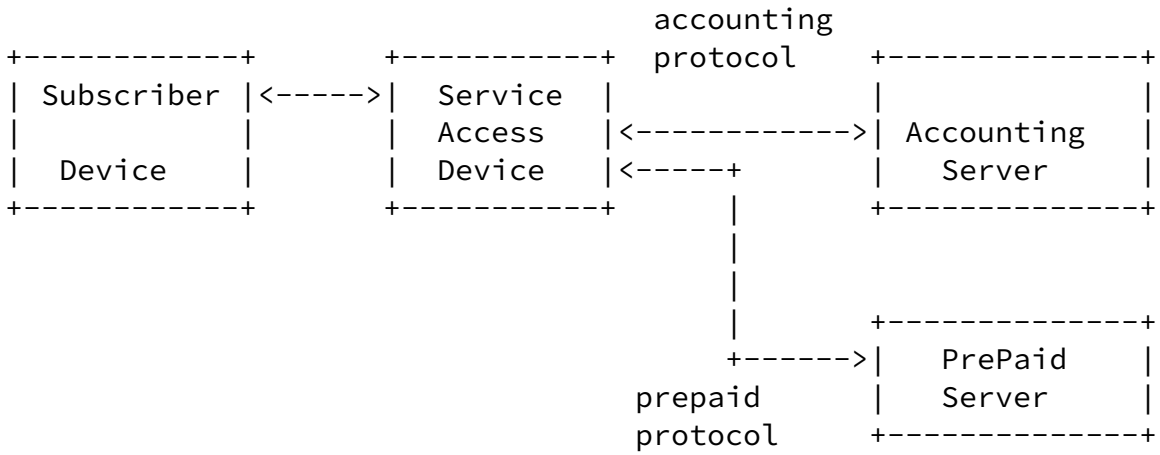


Figure 1 Basic Prepaid Architecture

The prepaid server and accounting server in this architecture model are logical entities. The real configuration MAY combine them into a single host.

There MAY exist protocol transparent RADIUS Proxies between prepaid client and prepaid server. These proxies transparently support the prepaid RADIUS extensions.

In order to generalize the solution, in this paper we generalize the Service Access Devices, which in reality may be a NAS in Dialup deployments, PDSN (Packet Data Serving Node) or HA (Home Agent) in CDMA2000 deployments, an 802.11 WLAN Access Points or GGSN (Gateway GPRS Serving Node) in GPRS/UMTS deployments. To actively participate in Prepaid procedures outlined here, the Service Access Device MUST have the Prepaid Client capabilities. Prepaid Client Capabilities include the ability to meter the usage for a prepaid data session; this usage includes time or volume (e.g. number of bytes) usage.

In the case of roaming scenarios using mobile IP (in a wireless or wireline network), the prepaid client functionality may be delegated to the Home Agent. It may also be possible to deliver limited prepaid services using RADIUS capabilities specified in RFC2865 and RFC2866.

Furthermore, the device including the prepaid client functionality may also have Dynamic Session Capabilities that include the ability to terminate a data session and/or change the filters associated

with a specific data session by processing Disconnect Messages and Change of Authorization messages as per [RFC3576].

In this document RADIUS is used as the AAA server. There are three kinds or categories of AAA servers. The AAA server in the home network, the HAAA, is responsible for authentication of the subscriber and also authorization of the service. In addition, the HAAA communicates with the Prepaid servers using the RADIUS protocol to authorize prepaid subscribers. In AAA based roaming deployments the AAA server in the visited network, the VAAA, is responsible for forwarding the RADIUS messages to the HAAA. The VAAA may also modify the messages. In roaming deployments, the visited network may be separated from the home network by one or more broker networks. The AAA servers in the broker networks, BAAA are responsible to route the RADIUS packets transparently and hence donÆt play an active roll in the Prepaid Data Service delivery.

In this document the Prepaid Server is described in functional terms related to their interface with the HAAA. The Prepaid Server interfaces to entities which:

i) Keep the accounting state of the prepaid subscribers (balance manager);
ii) Allow access service requests to be rated in real-time (Rating Engine); and
iii) Allow quota to be managed for a particular pre-paid service (Quota Server).

The various deployments for Prepaid are presented in the remainder of this section. The first deployment is the basic Prepaid data service and is depicted in figure 2. Here the Service Access Device which supports the prepaid client functionality, the HAAA and the Prepaid Server are collocated in the same provider network.

The Subscriber Device establishes a connection with one of several Access Devices in the network. The Service Access Device communicates with one or more HAAA servers in the network. To provide redundancy more than one HAAA may be available to use by a Service Access Device.

The network will have one or more Prepaid Servers. Multiple Prepaid Servers may be used to provide redundancy and load sharing. The interface between the HAAA and the PPS is implemented using the

RADIUS protocol in this specification. However, in cases where the PPS does not implement the RADIUS protocol, the implementation would have to map the requirements defined in this document to whatever protocol is used between the HAAA and the PPS.

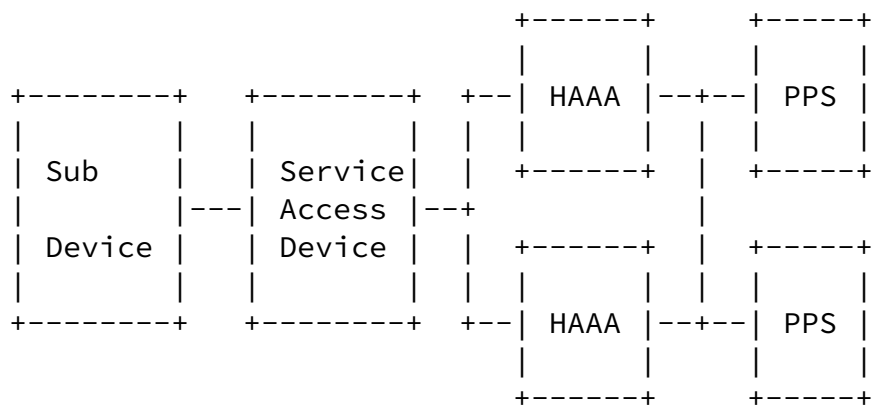


Figure 2 Basic Prepaid Access Architecture

Figure 3 shows a static roaming prepaid architecture that is typical of a wholesale scenario for Dial-Up users or a broker scenario used in Dial-Up or WLAN roaming scenarios.

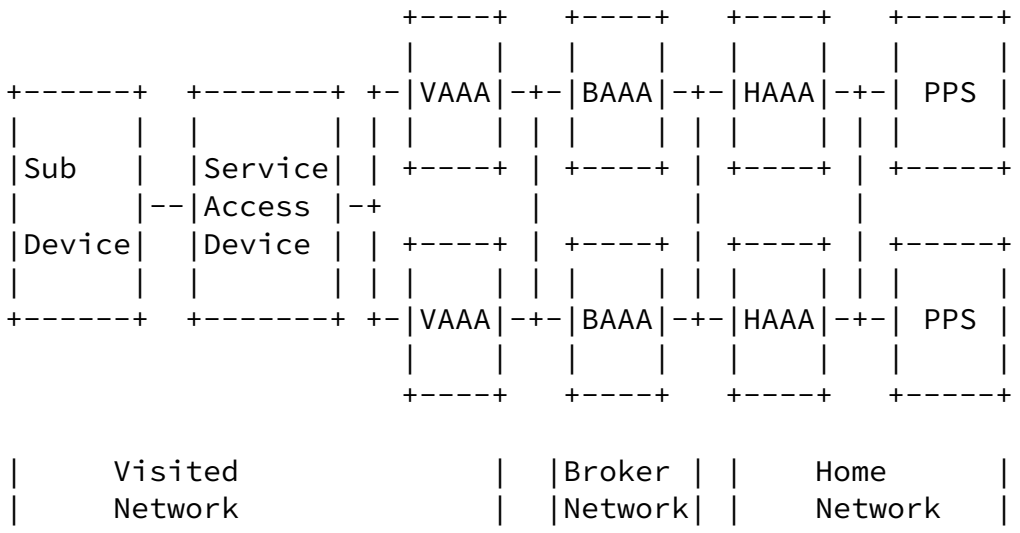


Figure 3 Static Roaming Prepaid Architecture

As in the basic prepaid architecture the subscriberÆs device establishes a connection with the Service Access Device (NAS, WLAN Access Point). The Service Access Device communicates with the

Visiting AAA server (VAAA) using the RADIUS protocol. Again for redundancy there maybe more then one VAAA. The VAAA communicate using the RADIUS protocol with AAA servers in the broker network (BAAA). There maybe more then one Broker Network between the Visited Network and the Home Network. The Home Network is the same as in the simple architecture.

To support dynamic roaming the network will utilize Mobile-Ip as illustrated in Figure 4. Note that typically the mobile device would be moving between networks that use the same technology such as Wireless or WLAN. Increasingly, device will be able to roam between networks that use different technology such as between WLAN and Wireless and Broadband. Fortunately, Mobile-Ip can address this type of roaming and therefore we need not be concerned with the underlying network technology.

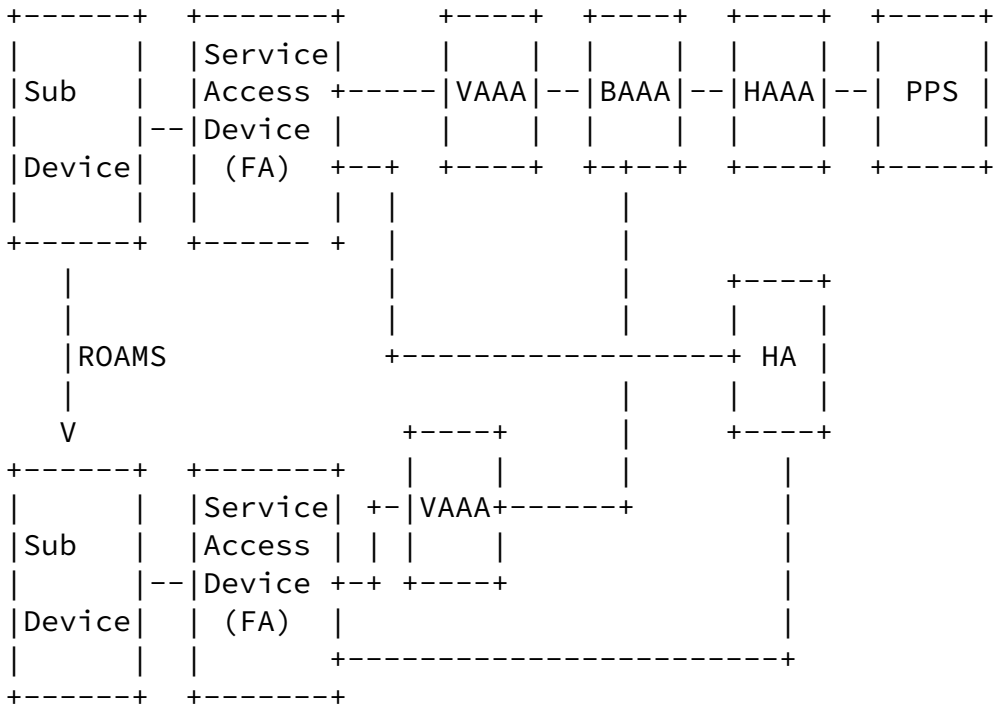


Figure 4 Roaming using Mobile-IP and pre-paid enabled Service Access Devices

In figure 4, the Subscriber device establishes a prepaid session between the Service Access Device in the foreign network, which has

prepaid capabilities. The subscriberÆs home address will be anchored at the Home Agent (HA) in the home network. The setup for this access service is identical to the cases covered above. Notice that the Service Access Device may be collocated with the Foreign Agent (FA) in case of Mobile-IPv4. As the subscriber device moves it establishes a connection with another Service Access Device in the same foreign network or in another foreign network. The prepaid data service should continue to be available. When a device associates to another Service Access Device it MUST re-authenticate at the new Service Access Device and de-associate or logoff from the old Service Access Device. Furthermore, any unused quota at the old Service Access Device MUST be promptly credited back to the subscribers account. The reason we say promptly, is because if the subscriber is very low on resources to start with, the subscriber may not have enough resources to log on to the new Service Access Device. The speed at which resources can be returned depend on the type of handoff procedure that is used. Some of the example of handoffs in wireless networks are dormant handoff, active handoff and fast handoff.

As well, notice that if the Service Access Devices could communicate with each other then there could be a way to accelerate a faster handoff procedure. In particular, it could accelerate the return of the unused portion of the quotas from the old Access Device.

Unfortunately, standards with regards to handoff are evolving with each network technology creating their own scheme to make the handoff procedures more efficient.

2.1 Why not existing RADIUS attributes?

It has been asked ôWhy not use existing RADIUS attributes to build a prepaid solution? This will allow us to have a solution with existing devices without code modification.ö

It is possible to build a prepaid solution using existing RADIUS attributes. The RADIUS server can simply send an Access-Accept message containing Session-Timeout(27) and set Termination-Action(29) to RADIUS-request. Upon receiving the Access-Accept message, the NAS will meter the duration of the session and upon termination of the session the NAS generate an Access-Request message again. The RADIUS server would re-authenticate the session

and reply with an Access-Accept message with additional time in Session-Timeout(27) or an Access-Reject message if there were no more resources in the userÆs account.

If the user terminates the session before the time expressed in Session-Timeout(27). The NAS will recover any unused time from the accounting stream.

There are several problems with such a solution:

-It only allows for time-based prepaid. The solution presented in this document allows for both time and volume based prepaid. As well as extensibility for other features such as tarified based solutions.

-Using accounting messages to recoup unused time may be problematic because RADIUS accounting messages are not real-time. A RADIUS server may store-and-forward accounting messages in batches. The solution presented in this paper does not rely on Accounting Packets at all. It uses Access-Request, messages which do flow through any network in real-time. Delaying accounting messages may cause revenue leakage.

-Session-Timeout(27) is not a mandatory attribute. If a prepaid subscriber is being serviced by a NAS that does not adhere to Session-Timeout then that subscriber will obtain unlimited service.

-Termination-Action(29) presents its own issues. First the behaviour of Termination-Action(29) is not mandatory. Second, according to RFC2865 Termination-Action fires when the Service is complete. But we should not be terminating the service while negotiating additional quota. The refreshing of the time quota should be transparent to the user. Because Termination-Action occurs when the Service is complete it is unclear whether or not the user experience would be transparent. For example, will the RADIUS server allocate the subscriber a new IP address? Furthermore, the RADIUS server has no way of telling why the Access-Request message was generated. The RADIUS server will have to wait for the corresponding accounting packet to determine the reason for this Access-Request message. Lastly re-authenticating the subscriber may take far too long. The solution presented in this document allows quota replenishing to occur in an undisruptive manner from the

perspective of the user. No re-authentication is required and quotas can be negotiated prior to the quotas running out.

-Prepaid ambiguity. Implementing prepaid using existing RADIUS attributes presents another problem. Due to the fact that the standard RADIUS attributes are not mandatory, then the correct prepaid operation is really an act of faith on the part of the RADIUS server. If Session-Timeout(27) and/or Termination-Action(29) are not supported, the prepaid subscriber will get free access. The solution described in this document, requires that a prepaid capable Service Access Device inform the RADIUS server whether or not it supports prepaid capabilities. The RADIUS server can now determine whether service should be granted or not. For example, if a prepaid subscriber is connected to a NAS that does not support prepaid, the RADIUS server can either instruct the NAS to tunnel the traffic to another entity in the home network that does support prepaid client function (e.g. Home Agent) or it may allow the subscriber to get access but restrict the traffic.

The prepaid solution we present is a robust carrier grade prepaid solution. It only requires the support of 2 mandatory attributes and one optional attribute. Furthermore, it does not really require much code support at the NAS. NASes already support measurement of time and volume. This solution requires that they advertise their prepaid capabilities in an Access-Request; that they generate an Access-Request Authorize-Only packet to obtain more quota at or before the quota is used up. It also requires that the NAS send an Access-Request with Authorize-Only when the session terminates to return any unused quota to the prepaid system.

Lastly the solution provided in this document is extensible. This document defines the basic exchanges between a prepaid capable NAS and a RADIUS server. The protocol can easily be extended to support tariff switching and other prepaid business models.

3. Use-cases

In this section we present a set of use cases that will help establish the requirements needed to deliver PrePaid data services. These use cases donÆt address how the PrePaid account is established or maintained. It is assumed that the PrePaid subscriber has obtained a valid account from a service provider such as a wireless operator or a WLAN operator.

To make the document as general as possible, the use cases cover the experience from the Service Access Device and not from the UserÆs Device. The connection between the UserÆs Device, which typically involves setting up a layer 2 session, e.g., PPP session or GPRS PDP Context, is specific to a given network technology and the details are not required to deliver a PrePaid service.

3.1 Simple pre-paid access use-case

A PrePaid subscriber connects to his home network. As usual, the Access Device that is servicing the subscriber will use the AAA infrastructure to authenticate and authorize the subscriber.

The Service Access Device sends a RADIUS Access-Request to the AAA system to authenticate the subscriber, and identify and authorize the service. The Access-Request includes the subscriberÆs credentials and may include the PrePaid capabilities of the Service Access Device. PrePaid capabilities MUST be included if the Service Access Device supports PrePaid functionality.

The AAA System proceeds with the authentication procedure. This may involve several transactions such as in EAP [RFC2284]. Once the subscriber has been authenticated, the AAA system determines that the subscriber is a PrePaid subscriber and requests that the PrePaid System authorize the PrePaid subscriber. The request MUST include the PrePaid Capabilities of the serving Service Access Device.

The PrePaid System will validate that the subscriber has a PrePaid Account; it will validate that the account is active; and will validate that the Service Access Device has the appropriate PrePaid capabilities. If all is in order, the PrePaid System will authorize the subscriber to use the network. Otherwise it will reject the request. The response is sent back to the AAA System. The response includes attributes to indicate the allocation of a portion of the subscriberÆs account called the initial quota (in units of time or volume) and optionally a threshold value.

The reason we allocate a portion of the userÆs account is that the user may be engaged in other Services that may draw on the same Prepaid account. For example the user may be engaged in a data session and a voice session. Although, these two services would

draw from the same account the involved separate parts of the system. If the entire quota was allocated to the data session then the user would have no more funds for a voice session.

The AAA system incorporates the PrePaid attributes received from the PrePaid System into an Access-Accept message that it sends back to the Service Access Device. Note the AAA System is responsible for authorizing the service whereas the PrePaid System is responsible for PrePaid authorization.

Upon receiving the Access-Response, the Service Access Device allows the PrePaid data session to start and it starts to meter the session based on time or volume, as indicated in the returned Quota

Once the usage for the session approaches the allotted quota (as expressed by the threshold), the Service Access Device will request an additional quota. The re-authorization for additional quota flows through the AAA system to the PrePaid System. The PrePaid System revalidates the subscriberÆs account; it will subtract the previous quota allocation from the userÆs account balance and if there is a balance remaining it will reauthorize the request with an additional quota allotment. Otherwise, the PrePaid System will reject the request. Note the replenishing of the quotas is a re-authorization procedure and does not involve re-authentication of the subscriber.

It is important to note that the PrePaid System is maintaining session state for the subscriber. This state includes how much account balance was allocated during the last quota allocation for a particular session and how much is left in the account. Therefore, it is required that all subsequent messages about the PrePaid session reach the correct PrePaid System.

Upon receiving a re-allotment of the quota, the Service Access Device will, continue the data service session until the new threshold is reached. If the request for additional quota cannot be fulfilled then the Service Access Device will let the subscriber use up the remaining quota and terminate the session.

Alternatively, instead of terminating the session, the Service Access Device may restrict the data session such that the subscriber can only reach a particular web server. This web server maybe used to allow the subscriber to replenish their account. This

restriction can also be used to allow new subscribers to purchase their initial PrePaid Service.

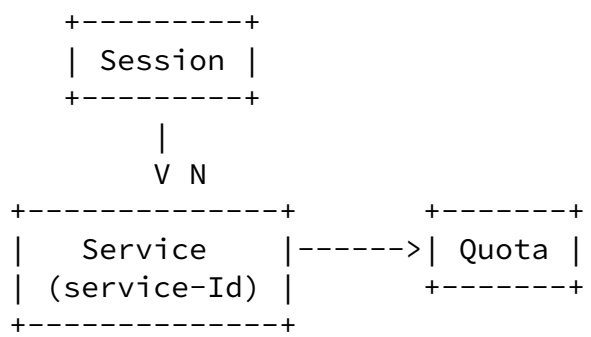
Should the subscriber terminate the session before the quota is used up, the remaining balance allotted to the session must be credited back to the subscriberÆs account.

As well, while the Access Device is waiting for the initial quota, the subscriber may have dropped the session. The initial quota must be credited back to the subscribers account.

3.2 Support for Multi-Services

Up to now we were looking at session that consisted of a single service, ôAccess Serviceö. An ôAccess Serviceö is the basic service that is provided to the user by the Service Access Device after successful authentication and authorization. When we donÆt differentiate between different types of services the ôAccess Serviceö aggregates all the services that the user my be engaged in on a particular Service Access Device. For example, the user may be browsing the web, and participating in a VoIP conversation, watching streaming video and downloading a file.

Some operators may want to distinguish these Services. Some services are billed at different rates and Services maybe metered differently. Therefore, the prepaid solution needs to be able to distinguish Services, and allocate quotas to the Services using different units (e.g. time, volume) and allow for those quotas to be utilized at different rates.

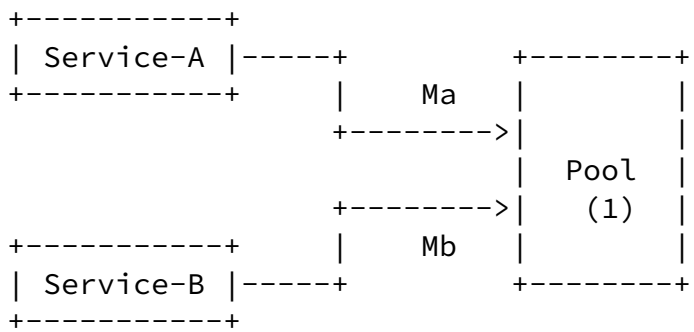


As shown in the above diagram, a Session can have N Services. Each service is identified by a Service-Id. The format of the Service-Id is not in the scope of this document but the Service-Id could be expressed as an IP flow using the IP 5-tuple (Source-IP and Port, the Destination-IP and Port, and the protocol). Each Service is allocated a Quota appropriate to the service.

3.3 Resource Pools

When working with multiple services which results in multiple quota allocation another problem arises. Even though quotas are portioned out in fractional parts of the users prepaid account, there could be a situation where one Service utilizes its quota faster then another Service. When the userÆs account is used up, there could be a situation where one Service is unable to obtain additional quota while another Service has plenty of quota remaining. Unless the quotas can be rebalanced, the Service Access Device would then have to terminate that Service. As well, even before that happens, the existence of several Services could generate an excessive amount of traffic as the services update their quotas.

One method to solve these problems is to utilize resource pools. Resource pools allow us to allocate resources to several services of a session by allocating resources to a pool and have services draw their quota from the pool at a rate appropriate to that service. When the quota allocated to the pool runs out, we replenish the pool.



As the above figure shows, Service-A and Service-B is bound to Pool(1). Ma and Mb are the pool multipliers (that are associated with Service-A and Service-B respectively) that determines the rate at which Service-A and Service-B draw from the pool.

The pool is initialized by taking the quota allocated to each service and multiplying it by Mn. Therefore, the amount of resources allocated to a pool is given by:

Poolr = Ma*Qa + Mb*Qb + . . .

A Pool is empty if:

Poolr <= Ca*Ma + Cb*Mb + . . .

where:
Ca,Cb are the consumed resources of Service-A and Service-B respectively.

Note that the resources assigned to the pool are unit less. That is, Service-A can be rated at $1 per Mbyte and Service-B can rated at $0.10 per Minute. In this case if we allocate $5 worth of resources on behalf of service-A to the pool we would set Ma = 10 and place 50 units into the pool. If we allocate $5 on behalf of Service-B to the Pool, then M=1 and place 50 units into the Pool. The pool would have a total sum of 100 units to be shared between the two services. Each Mbyte used by Service-A will draw 10 units from the pool and each minute used by Service-B will draw 1 unit from the pool.

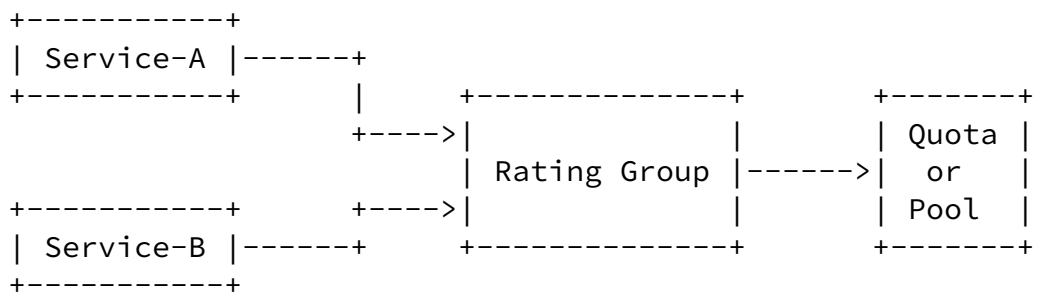
3.4 Support for Complex Rating Functions

The rate of use of a resource by a service can be very complex. Some services use resources (e.g. time, volume) linearly. For example, a service maybe consuming resources at a rate of $1 per Mbyte.

In some cases an operator may wish to apply a much more complex rating function. For example, a service provider may wish to rate a service such that the first N Mbytes are free, then the next M Mbytes are rated at $1 per Mbyte and volume above M bytes be rated at $0.50 per Mbyte. This rating function could be achieved by repeated message exchanges with the Prepaid System.

To avert the need to exchange many messages and to support even more complex rating functions we support Rating Groups. A Rating Group

is provisioned at the Service Access Device. As illustrated in the figure below, a Rating Group is associated with one or more Services and defines the rate that the services associated with the Rating Group consume the quota.



During authorization of the of a service, if the service is associated with a Rating Group, the Prepaid Client sends the Rating Group to the Prepaid Server. The prepaid service authorizes the Rating Group by assigning it a Quota and optionally assigning it to a Resource Pool.

When service that belongs to an authorized Rating Group is instantiated, then the Prepaid Client does not need to authorize that service. This could greatly reduce the amount of traffic between the Prepaid Client and the Prepaid Server.

3.5 Support for Roaming

For some networks it is essential that PrePaid Data Services be offered to roaming subscribers. Support for static and dynamic roaming models are needed. Static roaming is where the subscriber logs onto a foreign network. The foreign network has a roaming agreement directly with the home network or through a broker network or networks. The subscriber remains logged into the network until the subscriber changes location. When changing location a new connection and a new login procedure is required.

Dynamic roaming allows to subscriber to move between networks while maintaining a connection with the home network seamlessly. As the subscriber moves between networks, the data session is handed off between the networks.

In both roaming scenarios, the subscriber always authenticates with the home network. PrePaid authorization and quota replenishing for the session need to be received at the home network and more specifically at the PrePaid System where state is being maintained.

Dynamic roaming is particularly challenging. A subscriber that established a PrePaid Data Session may roam to another Access Device that doesnÆt not support PrePaid functionality. The system should be capable to continue the PrePaid session.

3.6 PrePaid termination

When fraud is detected by the PrePaid System, or when an error is detected, it may be beneficial for the PrePaid system to terminate a specific session for the subscriber or all the sessions of a subscriber.

Some errors can occur such that the PrePaid System is in a state where it is not sure whether the session is in progress or not. Under conditions such as this, the PrePaid system may wish to terminate the PrePaid data session to make sure that resources are not being utilized for which it canÆt charge for reliably.

Some handoff procedure used during dynamic roaming may require that the PrePaid system explicitly terminate the subscribers PrePaid data session at an Service Access Device. For example, if time based PrePaid service is being used and the mobile subscriber performs a dormant handoff, the PrePaid System needs to explicitly terminate the PrePaid session at the old Service Access Device.

4. Operations

4.1 General Requirements

4.1.1 Broker AAA Requirements

Broker AAA servers MUST support the Message-Authenticator(80) attribute as defined in [RFC2869]. If BAAA servers are used, the BAAA servers function is to forward the RADIUS packets as usual to the appropriate RADIUS servers.

Accounting messages are not needed to deliver a PrePaid service. However, accounting messages can be used to keep the PrePaid Server current as to what is happening with the PrePaid data session. Therefore, BAAA SHOULD deliver RADIUS Accounting messages using the pass through mode described in [RFC2866].

4.2 Authentication and Authorization for Prepaid Enabled Service Access Devices

The Service Access Device initiates the authentication and authorization procedure by sending a RADIUS Access-Request to the HAAA.

If the Service Access Device has PrePaid Client capabilities, it MUST include the PPAC(TBD) attribute in the RADIUS Access-Request. The PPAC(TBD) attribute indicates to the PrePaid server the PrePaid capabilities possessed by the Service Access Device. These are required in order to complete the PrePaid authorization procedures.

If the Service Access Device supports the Disconnect-Message or the Change-of-Authorization capabilities, then it SHOULD include the Dynamic-Capabilities attribute.

In certain deployments, there may be other ways in which to terminate a data session, or change authorization of an active session. For example, some Service Access Devices provide a session termination service via Telnet or SNMP. In these cases, the AAA server MAY add the Dynamic-Capabilities message to the Access-Request. Upon receiving the Change-of-Authorization message, the AAA server would then be responsible for terminating the session using whatever means that are supported by the device.

If the authentication procedure involves multiple Access-Requests (as in EAP), the Service Access Device MUST include the PPAC(TBD) attribute and the Dynamic-Capabilities attribute (if used) in at least the last Access-Request of the authentication procedure.

The Access-Request will be sent as usual to the HAAA. The packet may be proxied through zero or more BAAA.

Once the Access-Request arrives at the HAAA, the HAAA will authenticate the subscriber. If the subscriber is cannot be authenticated, the HAAA will send an Access-Reject message back to

the client. If the subscriber is authenticated, the HAAA will determine whether or not the subscriber is a PrePaid subscriber. The techniques used to determine whether or not a subscriber is a PrePaid subscriber is beyond the scope of this document. If the subscriber is not a PrePaid subscriber, then the HAAA will respond as usual with an Access-Accept or Access-Reject message. If the subscriber is a PrePaid Subscriber the HAAA SHALL forward the Access-Request to a PrePaid server for further authorization.

The Access-Request will contain the PPAC(TBD) attribute, the Dynamic-Capabilities attribute if one was included; the User-Name(1) attribute MAY be set to a value that would represent the SubscriberÆs PrePaid Identity. This attribute is used by the PrePaid server to locate the PrePaid SubscriberÆs account. For added security, the HAAA MAY also set the User-Password(2) attribute to the password used between the HAAA and the PrePaid server.

The PrePaid server lookups the subscriberÆs PrePaid account and will authorize the subscriber taking into consideration the Service Access Device PrePaid Client Capabilities.

Upon successful authorization, the PrePaid server will generate an Access-Accept containing the PPAC(TBD) attribute and the PPAQ(TBD) attribute.

The PPAC attribute returned to the client indicates the type of prepaid service to be provided for the session. The PPAQ(TBD) attribute includes:

- The QUOTA-Id, which is set by the PrePaid server to a unique value that is used to correlate subsequent quota requests;

- Volume and/or Time quotas, which are set to a value representing a portion of the subscribers account;

- MAY contain a Time or Volume Threshold that controls when the Service Access Device requests additional quota;

- The IP address of the Serving PrePaid Server and one or more alternative PrePaid Servers. This is used by the HAAA to route subsequent quota replenishing messages to the appropriate PrePaid server(s).

Note: Idle-Timeout(28) can be used to trigger the premature termination of a pre-paid service following subscriber inactivity.

Depending on site policies, upon unsuccessful authorization, the PrePaid server will generate an Access-Reject to terminate the session immediately. Alternatively, the PrePaid server may generate an Access-Accept blocking some or all of the traffic and/or redirect some or all of the traffic to a location where the subscriber can replenish their account for a period of time. Blocking of traffic is achieved by either Filter-Id(11) or NAS-Filter-Rule(see Redirect I-d). Redirection is achieved by sending Redirect-Id or Redirect-Rule defined in the Redirect I-d. The period of time before the blocked/redirected session last can be specified by Session-Timeout(27) attribute.

Upon receiving the Access-Accept from the PrePaid Server, the HAAA will append the usual service attributes and forward the packet to the Service Access Device. The HAAA SHOULD NOT overwrite any attributes already set by the PrePaid server. If the HAAA, receives an Access-Reject message, it will simply forward the packet to its client. Depending on site policies, if the HAAA fails to receive an Access-Accept or Access-Reject message from the PrePaid server it MAY do nothing or send an Access-Reject or an Access-Accept message back to its client.

4.3 Session Start Operation

The real start of the session is indicated by the arrival of Accounting-Request(Start) packet. The Accounting-Request (Start) MAY be routed to the PrePaid Server so that it can confirm the initial quota allocation.

Note that the PrePaid Server role is not to record accounting messages and therefore it SHOULD not respond with an Accounting Response packet.

If the Prepaid server does not receive the Accounting-Request(start) message it will only know that the session has started upon the first reception of a quota replenishment operation.

If the Prepaid server does not receive indication directly (via Accounting-Request(start)) or indirectly, it SHOULD after some

configurable time, deduce that the Session has not started. If the Service Access Device supports termination capabilities, the PPS SHOULD send a Disconnect Message to the Service Access Device to ensure that the session is indeed dead.

4.4 Mid-Session Operation

During the lifetime of a PrePaid data session the Service Access Device will request to replenish the quotas using Authorize-Only Access-Request messages.

Once the allocated quota has been reached or the threshold has been reached, the Service Access Device MUST send an Access-Request with Service-Type(6) set to a value of ôAuthorize Onlyö and the PPAQ(TBD) attribute.

The Service Access Device MUST also include NAS identifiers, and Session identifier attributes in the Authorize Only Access-Request. The Session Identifier should be the same as those used during the Access-Request. For example, if the User-Name(1) attribute was used in the Access-Request it MUST be included in the Authorize Only Access-Request especially if the User-Name(1) attribute is used to route the Access-Request to the Home AAA server.

The Authorize Only Access-Request MUST not include either User Password or Chap Password. In order to authenticate the message, the Service Access Device MUST include the Message-Authenticator(80) attribute. The Service Access Device will compute the value for the Message-Authenticator based on [RFC2869].

When the HAAA receives the Authorize-Only Access-Request that contains a PPAQ(TBD), it SHALL validate the message using the Message-Authenticator(80) as per [RFC2869]. If the HAAA receives an Authorize Only Access-Request that contains a PPAQ(TBD) but not a Message-Authenticator(80) it SHALL silently discard the message. An Authorize Only Access-Request message that does not contain a PPAQ(TBD) is either in error or belongs to another application (for example, a Change of Authorization message [RFC3576]). In this case the Authorize Only Access-Request will either be silently discarded or handled by another application (not in scope of this document).

Once the Authorize Only Access-Request message is validated, the HAAA SHALL forward the Authorize Only Access-Request to the

appropriate PrePaid Server. The HAAA MUST forward the Authorize Only Access-Request to the PrePaid server specified in the PPAQ(TBD). The HAAA MUST sign the message using the Message-Authenticator(80) and the procedures in [RFC2869]. As with the Access-Request message, the HAAA MAY modify the User-Name(1) attribute to a value that represents the userÆs internal PrePaid account in the PrePaid server. Note the PrePaid server could use the Quota-ID sub-attribute contained within the PPAQ(TBD) to locate the user account.

Upon receiving the Authorize Only Access-Request containing a PPAQ(TBD) attribute, the PrePaid server MUST validate the Message-Authenticator(80) as prescribed in [RFC2869]. If the message is invalid, the PrePaid server MUST silently discard the message. If it received an Authorize Only Access-Request message that does not contain a PPAQ(TBD) it MUST silently discard the message.

The PrePaid server will lookup the PrePaid session by using the PrePaid Quota Id contained within the PPAQ(TBD). The PrePaid Server would, take the last allocated quota and subtract that from the UserÆs balance. If there is remaining balance, the PrePaid server re-authorizes the PrePaid session by allocate an additional quota. The PrePaid server may want to calculate a different threshold values as well.

Upon successful re-authorization, the PrePaid server will generate an Access-Accept containing the PPAQ(TBD) attribute. The Access-Accept message MAY contain Service-Type(6) set to Authorize-Only and MAY contain the Message-Authenticator(80).

Depending on site policies, upon unsuccessful authorization, the PrePaid server will generate an Access-Reject or an Access-Accept with Filter-Id(11) or Ascend-Data-Filter (if supported) attribute and the Session-Timeout(27) attribute such that the PrePaid subscriber could get access to a restricted set of locations for a short duration to allow them to replenish their account, or create an account; or to browse free content.

Upon receiving the Access-Accept from the PrePaid server, the HAAA SHALL return the packet to its client. If the HAAA, receives an Access-Reject message, it will forward the packet. Depending on site policies, if the HAAA fails to receive an Access-Accept or an

Access-Reject message from the PrePaid server it MAY do nothing or it MAY send an Access-Reject message back to its client.

Upon receiving an Access-Accept, the Service Access Device SHALL update its quotas and threshold parameters with the values contained in the PPAQ(TBD) attribute. Note that the PrePaid server MAY update the PrePaidServer attribute(s) and these may have to be saved as well.

Upon receiving an Access-Accept message containing either Filter-Id(11) or Ascend-Data-Filter attributes, and or Session Timeout(27). The Service Access Device SHALL restrict the subscriber session accordingly.

4.5 Dynamic Operations

The PrePaid server may want to take advantage of the dynamic capabilities that are supported by the Service Access Device as advertised in the Dynamic-Capabilities attribute during the initial Access-Request.

There are two types of actions that the PrePaid server can perform: it can request that the session be terminated; or it can request that attributes associated with the session be modified. More specifically, it can modify previously sent PPAQ(TBD)

Both of these actions require that the session be uniquely identified at the Service Access Device. As a minimum the PrePaid server:

-MUST provide either the NAS-IP-Address(4) or NAS-Identifier(32)
-MUST provide at least one session identifier such as User-Name(1), Framed-IP-Address(), the Accounting-Session-Id(44).

Other attributes could be used to uniquely identify a PrePaid data session.

For a discussion on Dynamic Operations as they related Mutli-Service operations see further on.

4.5.1 Unsolicited Session Termination Operation

At anytime during a session the Prepaid Server may send a Disconnect Message to terminate a session. This capability is described in detail in [RFC3576]. The PrePaid server sends a Disconnect Message that MUST contain identifiers that uniquely identify the subscriberÆs data session and the Service Access Device servicing that session.

If the Service Access Device receives a Disconnect-Message, it will respond with either a Disconnect-ACK packet if it was able to terminate the session or else it will respond with a Disconnect-NAK packet.

Upon successful termination of a session the Service Access Device MUST return any unused quota to the Prepaid Server by issuing an Authorize Only Access-Request containing the PPAQ which contains any unused Quota and the Update-Reason set to ôRemote Forced Disconnectö.

4.5.2 Unsolicited Change of Authorization Operation

At anytime during the prepaid session the Prepaid Client may receive a Change of Authorization (CoA) message. A Prepaid Server may send a new Quota to either add additional quota or to remove quota already allocated for the service.

If the Change of Authorization contains a PPAQ then that PPAQ will override a previously received PPAQ. The PPAQ may contain more allocated Quota or less allocated quota. The PPS MUST NOT change the units used in the PPAQ.

If the newly received PPAQ reduces the amount of allocated quota beyond what is currently used then the Service Access Device will accept the new PPAQ and act as it normally would when the quota is used up. For example, if the threshold is reached then is request a quota update; if the quota received is less then the currently used level then the Service Access Device would follow the normal procedures followed when a quota is used up.

4.6 Termination Operation

The termination phase is initiated when either: the Subscriber logs off; the quotas have been consumed, or when the Service Access Device receives a Disconnect Message.

In the case where the user logged off, or the Service Access Device receives a Disconnect Message, the Service Access Device will send an Authorize-Only Access-Request message with a PPAQ(TBD) and Update-Reason attribute set to either ôClient Service terminationö or ôRemote Forced disconnectö and the currently used quota.

In the case where the quota has been reached, if the PPAQ(TBD) contained Termination-Action field, the Service Access Device will follow the specified action which would be to immediately terminate the service, to request more quota, or to Redirect/Filter the service.

4.7 Mobile IP Operations

In roaming scenarios using Mobile-IP, as the mobile subscriber roams between networks, or between different types of networks such as between WLAN and CDMA2000 networks, the PrePaid data session should be maintained transparently if the HA is acting as the Service Access Device.

As the subscriber device associates with the new Service Access Device (AP or PDSN that supports prepaid client capability), the Service Access Device sends a RADIUS Access-Request and the subscriber is re-authenticated and reauthorized. The Service Access Device MUST include the PPAC(TBD) attribute in the RADIUS Access-Request. In this manner the procedure follows the Authentication and Authorization procedure described earlier.

If the HA was acting as the Service Access Device before handoff, the userÆs prepaid session does not undergo any change after the handoff because the Mobile IP session is anchored at the HA and the userÆs Home IP address remains the same.

In the case of AP or PDSN acting as the Service Access Device it is likely that the userÆs IP address will change (Care of Address). Therefore, the ongoing prepaid session will have some impact. In the case the Service Access Device shall send an Access-Request. The Access-Request message is routed to the home network and MUST reach the PrePaid System that is serving the PrePaid session. The

PrePaid system will then correlate the new authorization request with the existing active session and will assign a quota to the new request. Any outstanding quota at the old Service Access Device MUST be returned to the PrePaid system. If the Mobile-IP nodes (HA and FA) supports registration revocation (Mobile IPv4 only). Specifically, the quota SHOULD be returned when the Service Access Device sends the Authorize Only Access-Request with PPAQ(TBD) Update-Reason set to either ôRemote Forced disconnectö or ôClient Service terminationö. In order to trigger the sending of this last Authorize Only Access-Request, the PrePaid system may issue a Disconnect Message [3576] to the Service Access Device.

If the subscriber has roamed to an Service Access Device that does not have any PrePaid Capabilities, PrePaid data service may still be possible by requesting the Home Agent (providing it has PrePaid Capabilities) to assume responsibilities for metering the service. The procedure for this scenario will be given in the next release of this draft.

4.8 Operation consideration for Multi-Services

This section describes the operation for supporting Prepaid for multi-services on the same Service Access Device. The operations for multi-services are very similar to operations for single service. Message flows illustrating the various interactions are presented at the end of this document.

A Service Access Device that supports prepaid operations for multi-services SHOULD set the ôMulti-Services Supportedö bit in the PPAC.

When working with multi-services, we need to differentiate between the services. A Service-Id attribute is used in the PPAQ(TBD) to uniquely differentiate between the services. The exact definition of the Service-Id attribute is out of scope for this document.

A PPAQ that contains a Service-Id is associated with that Service. A PPAQ that contains a Rating-Group-Id is associated with that Rating-Group. A PPAQ MUST not contain both a Rating-Group-Id and a Service-Id. A PPAQ that contains neither a Rating-Group-Id or a Service-Id applies to the ôAccess Serviceö.

4.8.1 Initial Quota Request

When operations with multi-services is desired, the Service Access Device will request the initial quota for the Service by sending a PPAQ containing the Service-Id for that Service in an Authorize-Only Access-Request packet. Similarly, if the Service Access Device supports Rating-Groups then it may request a prepaid quota for the Rating-Group by sending a PPAQ containing the Rating-Group-Id. In both cases the Update-Reason will be set to ôInitial-Requestö.

The Authorize-Only Access-Request packet may contain more than one PPAQ. The Authorize-Only Access-Request MUST include one or more attributes that serve to identify the session so that it can be linked to the original authentication. Which Session Identifier(s) is included is up to specific deployments. The Authorize-Only message must contain the Message-Authenticator(80) attribute for integrity protection of the Authorize-Only Access-Request message.

Upon receiving an Authorize-Only Access-Accept message containing one or more PPAQs the Prepaid System will allocate resources to each PPAQ. The resources, can be in units of time, volume as before. Each PPAQ will be assigned a unique QID that MUST appear in a subsequent PPAQ update for that service or rating-group. As well, the PPAQ MUST contain the Service-ID; or Group-ID; or neither, if the PPAQ applies to the ôAccess Serviceö.

4.8.2 Quota Update

Once the services start to utilize their allotted quota they will eventually need to replenish their quotas (either the threshold is reached or no more quota remains). To replenish the quota the Prepaid Client will send an Authorize-Only Access-Request message containing one or more PPAQs. Each PPAQ MUST contain the appropriate QID, Service-ID or Group-ID (or neither the Service-ID or Group-Id if the quota replenishment is for the ôAccess Serviceö). The Update-Reason filed will indicate either ôThreshold reachedö(3), or ôQuota reachedö(4). The Authorize-Only message must contain identifiers to identify the session.

Upon receiving an Authorize-Only Access-Request packet with one or more PPAQs the Prepaid Server will respond with a new PPAQ for that service. The PPAQ will contain a new QID, the Service-Id or Rating-Group-Id, a new Quota. If the Prepaid Server does not want to grant

additional quota to the Service it MUST include the Termination-Action subfield in the PPAQ that will instruct the Service Access Device what to do with the service.

4.8.3 Termination

When an allotted quota for the service is used up the Service Access Device shall act in accordance to the Termination-Action field set in the Quota. If the Termination-Action field is absent then the Service MUST be terminated.

If the Service is to be terminated then the Service Access Device shall send a PPAQ with the appropriate QID, the Service-Id, the used quota, and Update-Reason set to ôClient Service Terminationö.

If the ôAccess Serviceö has terminated, then all other services must be terminated as well. In this case the Service Access Device must report on all issued quotas for the various services. The Update-Reason field should be set to ôAccess Service Terminatedö.

Note when sending more then on PPAQ it may be required to send multiple Authorize Only Access-Requests.

4.8.4 Dynamic Operations

Dynamic operations for multi-services are similar to dynamic operations described for single service operations. The prepaid system may send a COA message containing a PPAQ for an existing service instance. The Service Access Device will match the PPAQ to the service using the Service-ID attribute. The new quota could be higher then the last allocated value or it could be lower. The Service Access Device must react to the new quota accordingly.

A Disconnect message may not be send for a specific service. A disconnect message terminates the ôAccess Serviceö. As such the Service Access Device must report back all unused quotas by sending an Authorize Only Access Request message containing a PPAQ for each active service. The Update-Reason shall indicate that the reason for the update reason.

4.8.5 Support for Resource Pools

If the Prepaid Client supports pools as indicated by setting the ôPools supportedö bit in the PPAC(TBD) then the Prepaid Server may associate a Quota with a Pool by including the Pool-Id and the Pool-Multiplier in the PPAQ(TBD).

When Resource Pools are used, the PPAQ must not use the threshold field.

4.8.6 Error Handling

If the Prepaid Server receives a PPAQ with an invalid QID it MUST ignore that PPAQ.

If the Prepaid Server receives a PPAQ containing a Service-Id, or a Rating-Group-Id that it does not recognize, then it MUST ignore that PPAQ.

If the Prepaid Client receives a PPAQ containing a Service-Id, or a Rating-Group-Id that it does not recognize, then it must ignore that PPAQ.

If the Prepaid Client receives a PPAQ that contains a Pool-Id without a Pool-Multiplier; or a Pool-Multiplier without a Pool-Id it must ignore that PPAQ.

4.9 Accounting Considerations

Accounting messages are not required to deliver PrePaid Data Service. Accounting message will typically be generated for PrePaid Data Service. This because accounting message are used for auditing purposes as well as for bill generation.

Accounting messages associated with PrePaid Data Sessions should include the PPAQ(TBD) attribute.

4.10 Service Access Device Operation

To be completed

4.11 Interoperability with Diameter Credit Control Application

RADIUS PrePaid solutions need to interoperate with Diameter protocol. Two possibilities exist: The AAA infrastructure is

Diameter based and the Service Access Device are RADIUS based; or the Service Access Device is Diameter based and the AAA infrastructure is RADIUS based.

The Diameter Credit Control Application [DIAMETERCC] describes how to implement a PrePaid using an all Diameter based infrastructure.

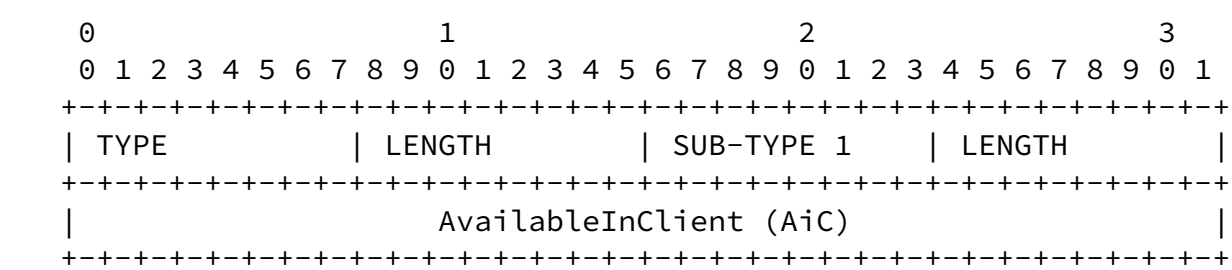
<This section to be completed.>

5. Attributes

This draft is using the RADIUS [RFC2865] namespace.

5.1 PPAC Attribute

The PrepaidAccountingCapability (PPAC) attribute is sent in the Access-Request message by a Prepaid Capable NAS and is used to describe the PrePaid capabilities of the NAS. The PPAC is available to be sent in an Access-Accept message by the Prepaid server to indicate the type of prepaid metering that is to be applied to this session.



TYPE : value of PPAC
LENGTH: 8
VALUE : String

The value MUST be encoded as follows:

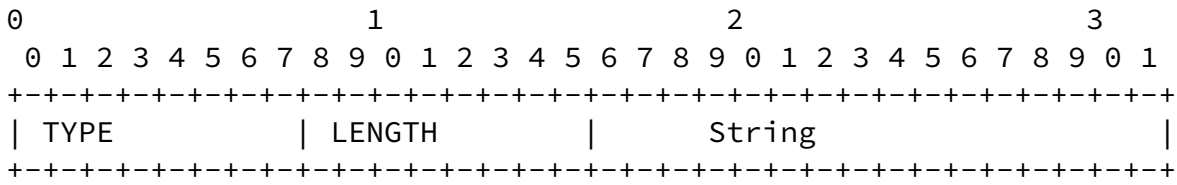
Sub-Type (=1) : Sub-Type for AvailableInClient attribute
Length : Length of AvailableInClient attribute (= 6 octets)
AvailableInClient (AiC):

The optional AvailableInClient Sub-Type, generated by the PrePaid client, indicates the PrePaid Accounting capabilities of the NAS and shall be bitmap encoded. The possible values are:

0x00000001	Volume metering supported.
0x00000002	Duration metering supported.
0x00000004	Resource metering supported.
0x00000008	Pools supported
0x00000010	Rating groups supported
0x00000020	Multi-Services supported.
Others	Reserved

5.2 Session Termination Capability

The value shall be bitmap encoded rather than a raw integer. This attribute shall be included RADIUS Access-Request message to the RADIUS server and indicates whether or not the NAS supports Dynamic Authorization.



Type : value of Session Termination Capability
Length: = 4
String encoded as follows:

0x00000001	Dynamic Authorization Extensions (rfc3576) is supported.

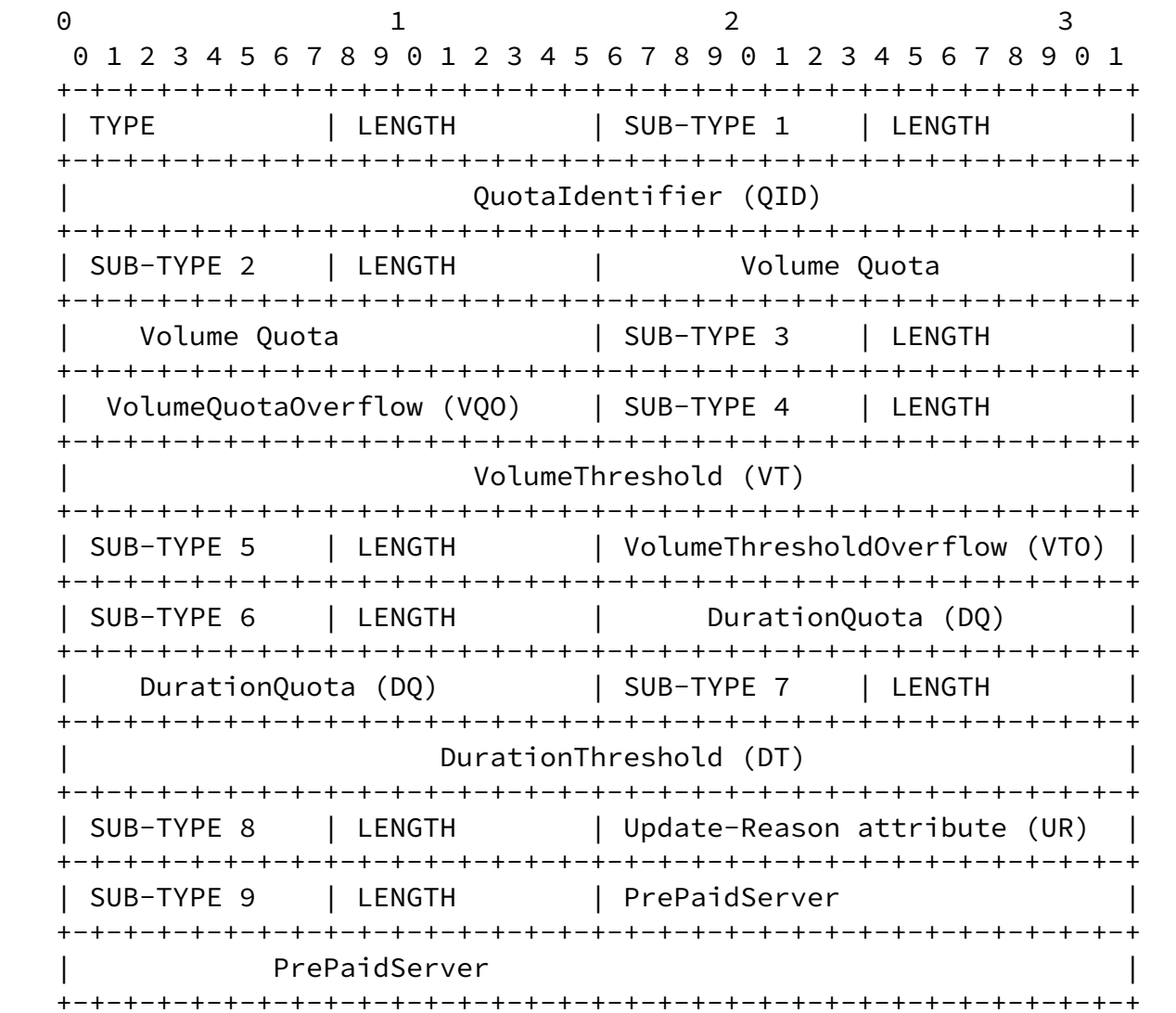
5.3 PPAQ Attribute

One or more PPAQ(TBD) attributes are available to be sent in Authorize Only Access-Request and Access-Accept messages. In Authorize Only Access-Request messages it is used to report usage and request further quota or request prepaid quota for a new service

instance; in an Access-Accept message it is used to allocate the quotas (initial quota and subsequent quotas).

When concurrent service are supported a PPAQ is associated with a specific service as indicated by the presence of Service-Id; or a Rating Group, as indicated by the presence of a Rating-Group-Id; or the ôAccess Serviceö as indicated by the absence of a Service-Id or a Rating-Group-Id.

The attribute consists of a number of subtypes. Subtypes not used are omitted in the message.



Type : Value of PPAQ
Length: variable, greater than 8

String: The String value MUST be encoded as follows:

Sub-Type (=1): Sub-Type for QuotaIDentifier attribute
Length : Length of QuotaIDentifier attribute (= 6 octets)

QuotaIDentifier (QID):

The QuotaIDentifier Sub-Type is generated by the PrePaid server at allocation of a Volume and/or Duration Quota. The on-line quota update RADIUS Access-Request message sent from the Service Access Device to the PPS shall include a previously received QuotaIDentifier.

Sub-Type (=2): Sub-Type for VolumeQuota attribute
Length : length of VolumeQuota attribute (= 6 octets)

VolumeQuota (VQ):

The optional VolumeQuota Sub-Type is only present if Volume Based charging is used. In RADIUS Access-Accept message (PPS to Service Access Device direction), it indicates the Volume (in octets) allocated for the session by the PrePaid server. In RADIUS Authorize Only Access-Request message (Service Access Device to PPS direction), it indicates the total used volume (in octets) for both forward and reverse traffic applicable to PrePaid accounting.

Sub-Type (=3): Sub-Type for VolumeQuotaOverflow
Length : length of VolumeQuotaOverflow attribute (= 4 octets)

VolumeQuotaOverflow (VQO):

The optional VolumeQuotaOverflow Sub-Type is used to indicate how many times the VolumeQuota counter has wrapped around 2^32 over the course of the service being provided.

Sub-Type (=4): Sub-Type for VolumeThreshold attribute
Length : length of VolumeThreshold attribute (= 6 octets)

VolumeThreshold (VT):

The VolumeThreshold Sub-Type shall always be present if VolumeQuota is present in a RADIUS Access-Accept message (PPS to Service Access Device direction). It is generated by the PrePaid server and indicates the volume (in octets) that shall be used before requesting quota update. This threshold should not be larger than the VolumeQuota.

Sub-Type (=5): Sub-Type for VolumeThresholdOverflow
Length : Length of VolumeThresholdOverflow attribute (= 4 octets)

VolumeThresholdOverflow (VTO):

The optional VolumeThresholdOverflow Sub-Type is used to indicate how many times the VolumeThreshold counter has wrapped around 2^32 over the course of the service being provided.

Sub-Type (=6): Sub-Type for DurationQuota attribute
Length : length of DurationQuota attribute (= 6 octets)

DurationQuota (DQ):

The optional DurationQuota Sub-Type is only present if Duration Based charging is used. In RADIUS Access-Accept message (PPS to Service Access Device direction), it indicates the Duration (in seconds) allocated for the session by the PrePaid server. In on-line RADIUS Access-Accept message (PPC to PPS direction), it indicates the total Duration (in seconds) since the start of the accounting session related to the QuotaID.

Sub-Type (=7): Sub-Type for DurationThreshold attribute
Length : length of DurationThreshold attribute (= 6 octets)

DurationThreshold (DT):

The DurationThreshold Sub-Type shall always be present if DurationQuota is present in a RADIUS Access-Accept message (PPS to Service Access Device direction). It represents the duration (in seconds) that shall be used by the session before requesting quota update. This threshold should not be larger than the DurationQuota and shall always be sent with the DurationQuota.

Sub-Type (=8): Sub-Type for Update-Reason attribute
Length : length of Update-Reason attribute (= 4 octets)

Update-Reason attribute (UR):

The Update-Reason Sub-Type shall be present in the on-line RADIUS Access-Request message (Service Access Device to PPS direction). It indicates the reason for initiating the on-line quota update operation. Update reasons 4, 5, 6, 7 and 8 indicate that the associated resources are released at the client side, and therefore the PPS shall not allocate a new quota in the RADIUS Access_Accept message.

1. Pre-initialization
2. Initial Request
3. Threshold Reached
4. Quota Reached
5. Remote Forced Disconnect
6. Client Service Termination
7. ôAccess Serviceö Terminated
8. Service not established

Sub-Type (=9) : Sub-Type for PrePaidServer attribute
Length : Length of PrePaidServer
(IPv4 = 6 octets, IPv6= 18 octets

PrePaidServer:

The optional, multi-value PrePaidServer indicates the address of the serving PrePaid System. If present, the Home RADIUS server uses this address to route the message to the serving PrePaid Server. The attribute may be sent by the Home RADIUS server. If present in the incoming RADIUS Access-Accept message, the PDSN shall send this attribute back without modifying it in the subsequent RADIUS Access-Request message, except for the first one. If multiple values are present, the PDSN shall not change the order of the attributes.

Sub-Type (=10) : Sub-Type for Service ID
Length : Length of Service ID

Service-Id:

Opaque string that uniquely describes a service instance for which we want to apply prepaid metering to. A Service-Id could be an IP 5-tuple (source address, source port, destination address, destination port, protocol). If Service-ID is present in the PPAQ the PPAQ applies to that Service. If a PPAQ does not contain a Service-Id then the PPAQ applies to the Access Service.

Sub-Type (=11) : Sub-Type for Rating-Group-Id
Length : 6

Rating-Group-Id

Identifies that this PPAQ is associated with resources allocated to a Rating Group with the corresponding ID.

Sub-Type (=12) : Sub-Type for Termination-Action
Length : 6

This field is an enumeration of the action to take when the prepaid server does not grant additional quota. Valid actions are as follows:

0 Reserved
1 Terminate
2 Request More Quota
3 Redirect/Filter

Sub-Type (=13) : Pool-Id
Length : 6

Identifies the Pool that this quota is to be associated with.

Sub-Type (=14) : Pool-Multiplier
Length : 6

The pool-multiplier determines the weight that resources are inserted into the pool and the rate at which resources are taken out of the pool by this Service, or Rating-Group.

NOTES:

Either Volume-Quota or Time-Quota MUST appear in the attribute.

Volume Threshold may only appear if Volume Quota appears

A PPAQ MUST NOT CONTAIN both a Service-Id and a Rating-Group-Id.

A PPAQ that does not contain a Service-ID or a Rating-Group-Id applies to the ôAccess Serviceö.
When the PPAQ contains a Pool-Id it MUST also contain the Pool-Multiplier.

5.4 Table of Attributes

TO BE COMPLETED.

Request Accept Reject Challenge # Attribute

Authorize_Only Request Accept Reject

6. Security Considerations

The protocol exchanges described are susceptible to the same vulnerabilities as RADIUS and it is recommended that IPsec be employed to afford better security.

If IPsec is not available the protocol in this draft improves the security of RADIUS. The various security enhancements are explained in the following sections.

6.1 Authentication and Authorization

RADIUS is susceptible to replay attacks during the Authentication and Authorization procedures. A successful replay of the initial Access-Request could result in an allocation of an initial quota.

To thwart such an attack...

6.2 Replenishing Procedure

A successful replay attacks of the Authorize Only Access-Request could deplete the subscribers prepaid account.

To be completed.

7. IANA Considerations

To be completed.

This draft does create RADIUS attributes. However, the authors recognize that it may not be possible to obtain such attributes. Therefore, in subsequent drafts it will be proposed to use a Vendor space as an Application Space.

8. Normative References

[RFC2026] Bradner, S., "The Internet Standards Process -- Revision 3", RFC 2026, October 1996.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, March 1997.

[RFC2865] Rigney, C., Rubens, A., Simpson, W. and S. Willens, "Remote Authentication Dial In User Server (RADIUS)", RFC 2865, June 2000.

[RFC2866] Rigney, C., "RADIUS Accounting", RFC 2866, June 2000.

[RFC2869] Rigney, C., Willats, W., Calhoun, P., "RADIUS Extensions", RFC 2869, June 2000.

[RFC2868] Zorn, G., Leifer, D., Rubens, A., Shriver, J., Holdrege, M., Goyret, I., "RADIUS Attributes for Tunnel Protocol Support" , RFC 2868, June 2000.

[RFC3576] Chiba, M., Dommety, G., Eklund, M., Mitton, D., Aboba, B., "Dynamic Authorization Extensions to Remote Authentication Dial-In User Service (RADIUS)", RFC 3576, February 2003.

[DIAMETERCC] Work in Progress.

[REDIRECT] RADIUS Redirection Internet Draft. Work in progress.

RFC 2284 EAP

9. Call Flows

This section includes call flows illustrating various scenarios enabled by this specification.
The following are used in the call flows:

RADIUS packets:

AR	Access Request
ARA	Access Accept
AC	Accounting Requests
A	Authorize-Only Access-Request
AA	Access-Accept for Authorize-Only Access-Request

RADIUS Attributes:

PPAQ	PPAQ as defined in this specification
SID	One or more attributes representing the Session that the RADIUS packets is correlated to.
PPAC	PPAC as defined in this specification
ASID	Acct-Session-Id as defined by RADIUS
MSID	Acct-Multi-Session-Id as define by RADIUS

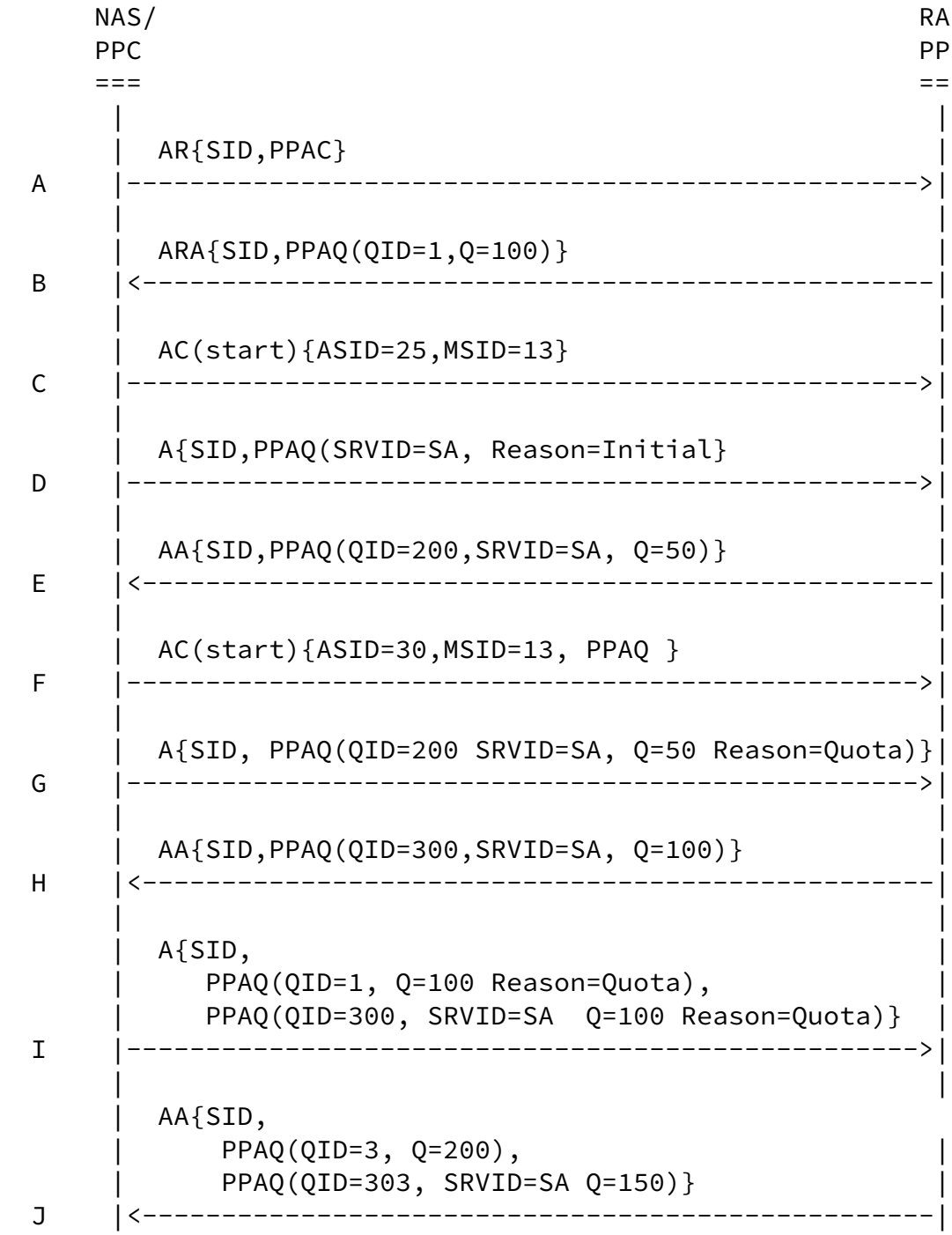
PPAQ fields:

SRVID	Service-Id
Reason	Update-Reason
QID	Quota-Id

9.1 Simple Concurrent Services

In this scenario the Prepaid Client authenticates and authorizes the user. The Prepaid Server responds back with Prepaid Quota for the ôAccess Serviceö instance. The NAS then request quota for Service-A.

Accounting is turned on.



A This is the initial Access-Request that indicates the Prepaid Capabilities of the NAS. In this scenario it will indicate

that Concurrent Session are supported. Access-Request also includes SID (Session Id) which is the Session Identifier assigned by this NAS to session. Session Identifier is out of scope in this document. It can be a single attribute such as 3GPP2 Correlation ID or it could be a set of attributes that define a session.

B RADIUS authenticates the user and determines that the user is prepaid. RADIUS responds with a PPAQ for the ôAccess Serviceö (PPAQ does not contain a Service-ID or Rating-Group-ID). The PPAQ has a QID=1 assigned by the Prepaid System and Quota of Q=100. The quota could be time or volume and may or may not have a threshold associated with it.

C NAS starts the Access Service and generates an Accounting-Request (Start) message as normal. It will include the Acct-Session-Id and may include the Acct-Multi-Session-Id.

D The NAS wants to start a new Service, call it Service-A. It sends an Authorize-Only access request to RADIUS. The SID links this Authorize-Only access request to the initial Authentication & Authorization (Step-A and Step-B).The Authorize-Only message contains a PPAQ requesting quota for Service-A, Update-Reason = Initial-Request.

E PPS checks the resources available to the user and assigns 50 units (time/volume etc) to this service. RADIUS sends an Access Accept message contain a PPAQ assigning quota Q=50 for Service-A. The PPAQ contains a QID = 200.

F NAS starts Service-A and sends an Accounting-Request (Start) message for that service. Acct-Multi-Session-Id can be used to tie all of the sessions in the accounting streams together.

G Quota for Service-A requires refreshing, the quota was completely used). An Authorize-Only message is sent containing a PPAQ with QID = 200 which corresponds to the prior QID received for this service. Note QID is sufficient for the PPS server to link this request to the previous request and hence to the original authentication steps. Therefore SID is not really required. The PPAQ will report the used part of the quota (50 units).

H RADIUS deducts the used quota from the users accounts and reserves 50 more additional units for a total quota of 100 (Q=100) for Service-A. It sends back a PPAQ with QID=300.

I NAS needs to refresh both the ôAccess Serviceö and Service-A. It sends an Authorize Only message contain two PPAQs, one for the Main Service with QID=1 and one for Service-A with QID=300. Each PPAQ reports the used resources so far and the

reason why the update is being sent.

J RADIUS responds back with two PPAQs. The PPAQ without the Service-Id grants an additional 100 units for a total of 200 units to the ôAccess Serviceö û QID=3; the other PPAQ, containing SRVID=SA grants an additional 50 units for a total quota to service-a of 150 units û QID=303.

This step illustrates why SRVID needs to be specified in the PPAQ. If it were not, then the NAS would not be able to differentiate between the PPAQs. QIDs are not sufficient to correlate the PPAQ to a service since they are changed (and not necessarily sequentially) by the PPS at every transaction.

In this scenario, notice how each PPAQ attribute represents a sequential conversation about a service between the Prepaid Client and the Prepaid Server. The links between the messages are the QIDs and the Service-Ids.

As well, notice how a SID is needed to tie the Authorize-Only messages to the Authentication steps. This SID is only really needed the first time a PPAQ is sent û since the PPAQ does not have a QID.

Accounting messages have an Accounting-Session-ID. But that is not enough to allow the back end system to associate that accounting message with a particular Service. We therefore need the PPAQ in the accounting message.

Acknowledgments

The authors would like to thank Mark Grayson (Cisco) and Nagi Jonnala for their contribution to this draft.

Author's Addresses

Avi Lior
Bridgewater Systems
303 Terry Fox Drive
Suite 100

Parviz Yegani, Ph.D.
Mobile Wireless Group
Cisco Systems
3625 Cisco Way

Ottawa Ontario
Canada
avi@bridgewatersystems.com

San Jose, CA 95134
USA
pyegani@cisco.com

Kuntal Chowdhury
Nortel Networks
2221, Lakeside Blvd,
Richardson, TX-75082
chowdury@nortelnetworks.com

Yong Li
Bridgewater Systems
303 Terry Fox Drive
Suite 100
Ottawa Ontario
Canada
Yong.li@bridgewatersystems.com

Intellectual Property Statement

Disclaimer of Validity

INTERNET ENGINEERING TASK FORCE DISCLAIM ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.

Copyright Statement

Expiration Date

This memo is filed as draft-lior-radius-extensions-for-prepaid-05.txt, and will expire 17 January, 2005.